REMARKS

Claim objections

Claim 14 has been objected to because it should depend from claim 13, instead of from claim 16 as originally presented. Applicant has corrected this error.

Claim 19 has been objected to because the phrase "that includes logic" better reads "including logic." Claim 19 has been amended in this suggested manner.

Claims 1-6

Claim 1 is an independent claim, from which claims 2-6 ultimately depend. Claims 1-6 have been rejected under 35 USC 102(b) as being anticipated by Kitamura (4,701,915). Applicant submits that at least as amended, claim 1 is patentable over Kitamura, such that claims 2-6 are patentable at least because they depend from a patentable base independent claim.

The element of claim 1 reciting "outputting the transaction from the pipeline into an error queue" has been amended in two ways. First, the transaction is output from the pipeline into the error queue "such that the transaction is output from the pipeline into the error queue *only in response to* the correctable error having been detected while processing the transaction within the pipeline." Applicant submits that this added language was at least implicit in the claim, and has been made explicit; i.e., there is no reason otherwise to output a transaction into an *error* queue unless an error has been first detected while processing the transaction. In any case, additional support for this limitation is found in the patent application as filed at least in FIG. 5, parts 506, 507, 508, 510, and 514; note particularly that the transaction is output from the pipeline to the error queue in part 514 only in response to an error being detected in part 507, as indicated by the decision box of part 508. Where an error is not detected in part 507, the decision box of part 508 causes the method 500 to perform part 510, where the transaction is not output from the pipeline into the error queue.

Second, this element of claim 1 has been amended so that "the error queue *only* stor[es] transactions within which correctable errors have been detected and *not* transactions within which correctable errors have not been detected." Applicant again submits that this added language was at least implicit in the claim, and has been made explicit; i.e., it stands to reason that an *error* queue only stores transactions that have errors and not transactions that do not have errors. In any case, additional support for this limitation is found in the patent application as filed, at least in FIG. 5, parts 507, 508, 510, and 514. Note particularly that the transaction is output from the pipeline for storage in the error queue in part 514 only after an error has been detected in part 507. That is, as indicated by the decision box of part 508, if an error is detected in the transaction, then the method 500 performs part 514, in which the transaction is stored in the error queue. If an error is not detected in the transaction, as also indicated by the decision box of part 508, then the method 500 instead does not perform part 514, but rather performs part 510, where the transaction is not stored in the error queue.

Applicant submits that at least these two added limitations of claim 1 are not disclosed by Kitamura. It is noted first that the Examiner has equated the data register 12 of Kitamura as the error queue of claim 1. (See, e.g., office action, p. 3, para. 3.) Next, Kitamura discusses its pipeline processing technique as follows.

> Describing, with reference to FIG. 1, the operation in the case of no error in the microinstruction read from the control storage 11, a microinstruction for pipeline processing read from the control storage 11 is written into the data register 12 and the parity check circuit 16, simultaneously. . . . .
>
> If no error is detected by the parity check circuit 16 or the error detection and correction circuit 14, the pipeline controller 15 executes a predetermined operation. . . . .
>
> Next, an explanation of the operation when an error exists in the microinstruction read from the control storage 11, and when an error correction code is used for the microinstruction are provided. . . . .
>
> . . . .
>
> When the microinstruction is read from the control storage 11 into the parity check circuit 16, the parity check circuit 16 . . . detects whether or not there is a bit error in the microinstruction by means of the parity check. . . . .

. . . .

[T]he error detection and correction circuit 14 corrects the bit error produced in the microinstruction in the data register 12 . . . .

(Col. 4, ll. 1-25 and ll. 50-53; col. 5, ll. 12-17.) Thus, Kitamura operates as follows. A microinstruction is read from the control storage 11 and written into the data register 12. If no error is detected in the microinstruction stored in the data register 12, then it is processed normally. If an error is detected in the microinstruction stored in the data register 12, then the circuit 14 corrects the error while the microinstruction remains in the data register 12.

Therefore, Kitamura does not anticipate claim 1 in at least the following two ways. First, the claimed invention is limited to the transaction being output from the pipeline into the error queue *only in response to* the error having been detected. By comparison, Kitamura outputs its microinstruction to the data register 12 *prior to* detecting whether there is an error in the microinstruction, and not *in response to* detect whether there is an error, as in claim 1. Furthermore, Kitamura *always* outputs the microinstruction into the data register 12, regardless of whether the microinstruction has an error or not. As such, Kitamura does not output the microinstruction into the data register 12 *only* in response to detecting whether there is an error in the microinstruction.

Second, the claimed invention is limited to the error queue *only* storing transactions within which errors have been detected *and not* transactions within which correctable errors have not been detected. By comparison, Kitamura stores within the data register 12 *both* microinstructions having errors *as well as* microinstructions that do *not* have errors. Therefore, Kitamura does not use its data register 12 to *only* store transactions that have errors *and not* transactions that do not have errors. For these reasons, then, Kitamura does not anticipate claim 1 at least as has been amended.

Claims 7-12

Claim 7 is an independent claim, from which claims 8-12 ultimately depend. Claims 7-12 have been rejected under 35 USC 103(a) as being unpatentable over Kitamura in view of Weber (6,631,448). Applicant submits that at least as has been amended, claim 7 is patentable over Kitamura in view of Weber, such that claims 8-12 are patentable at least because they depend from a patentable base independent claim.

Claim 7 has been amended in a similar manner as to which claim 1 has been amended. Thus, in claim 7, a transaction is output from the pipeline into the error queue *only in response to* a correctable error having been detected within the transaction while processing the transaction within the pipeline. Likewise, the error queue *only* stores transactions within which correctable errors have been detected and *not* transactions within which correctable errors have not been detected.

The Examiner has relied upon Kitamura as teaching all the limitations of claim 7 except for the limitations of claim 7 directed to a plurality of nodes interconnection to one another, where each node has a plurality of processors and local RAM, which the Examiner instead finds in Weber in rejecting claim 7 over Kitamura in view of Weber. However, as has been discussed in relation to claim 1 above, Kitamura does not disclose outputting a transaction into an error queue *only in response to* a correctable error having been detected, and Kitamura does not disclose the error queue *only* storing transactions within which errors have been detected and *not* transactions within which errors have not been detected. Weber also does not disclose these aspects of claim 7, in that it is specifically silent as to detecting and correcting errors within transactions. Because neither Kitamura nor Weber disclose these amended limitations of claim 7, not all the limitations of claim 7 are found in Kitamura in view of Weber, such that claim 7 as amended is non-obvious over Kitamura in view of Weber.

Claims 13-17

Claim 13 is an independent claim, from which claims 14-17 ultimately depend. Claims 13-17 have been rejected under 35 USC 102(b) as being anticipated by Kitamura. Applicant submits that at least as has been amended, claim 13 is patentable over Kitamura, such that claims 14-17 are patentable at least because they depend from a patentable base independent claim.

Claim 13 has been amended in a similar manner as to which claim 1 has been amended. Thus, in claim 13, a transaction is routed from the pipeline into the error queue *only in response to* a correctable error having been detected within the transaction while processing the transaction within the pipeline. Likewise, the error queue *only* stores transactions within which correctable errors have been detected and *not* transactions within which correctable errors have not been detected. As has been discussed in relation to claim 1 above, Kitamura does not disclose routing a transaction into an error queue *only in response to* a correctable error having been detected, and Kitamura does not disclose the error queue *only* storing transactions within which errors have been detected and *not* transactions within which errors have not been detected. Therefore, Kitamura does not anticipate claim 13.

Claim 18

Claim 18 is an independent claim, and has been rejected under 35 USC 102(b) as being anticipated by Kitamura. Applicant submits that at least as has been amended, claim 18 is patentable over Kitamura. Claim 18 has been amended in a similar manner as to which claim 1 has been amended. Thus, in claim 18, a transaction is routed from the pipeline into the error queue *only in response to* a correctable error having been detected within the transaction while processing the transaction within the pipeline. Likewise, the error queue *only* stores transactions within which correctable errors have been detected and *not* transactions within which correctable errors have not been detected. As has been discussed in relation to claim 1 above, Kitamura does not disclose routing a transaction into an error queue *only in response to* a correctable error

having been detected, and Kitamura does not disclose the error queue *only* storing transactions within which errors have been detected and *not* transactions within which errors have not been detected. Therefore, Kitamura does not anticipate claim 18.

Claim 19

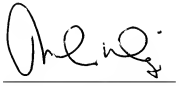
Claim 19 is an independent claim, and has been rejected under 35 USC 103(a) as being unpatentable over Kitamura in view of Weber. Applicant submits that at least as has been amended, claim 19 is patentable over Kitamura in view of Weber. Claim 19 has been amended in a similar manner as to which claim 1 has been amended. Thus, in claim 19, a transaction is output from the pipeline into the error queue *only in response to* a correctable error having been detected within the transaction while processing the transaction within the pipeline. Likewise, the error queue *only* stores transactions within which correctable errors have been detected and *not* transactions within which correctable errors have not been detected.

As has been discussed in relation to claim 1 above, Kitamura does not disclose outputting a transaction into an error queue *only in response to* a correctable error having been detected, and Kitamura does not disclose the error queue *only* storing transactions within which errors have been detected and *not* transactions within which errors have not been detected. Weber also does not disclose these aspects of claim 19, in that it is specifically silent as to detecting and correcting errors within transactions. Because neither Kitamura nor Weber disclose these amended limitations of claim 19, not all the limitations of claim 19 are found in Kitamura in view of Weber, such that claim 19 as amended is non-obvious over Kitamura in view of Weber.

Conclusion

Applicants have made a diligent effort to place the pending claims in condition for allowance, and request that they so be allowed. However, should there remain unresolved issues that require adverse action, it is respectfully requested that the Examiner telephone Mike Dryja, Applicants' Attorney, at 425-427-5094, so that such issues may be resolved as expeditiously as possible. For these reasons, this application is now considered to be in condition for allowance and such action is earnestly solicited.

Respectfully Submitted,



October 26, 2007
Date

Michael A. Dryja, Reg. No. 39,662
Attorney/Agent for Applicant(s)

Law Offices of Michael Dryja
1474 N Cooper Rd #105-248
Gilbert, AZ 85233
tel: 425-427-5094
fax: 425-563-2098